CS271 Spring 2021 Computer Graphics II

HomeWork 5

Name: 任怡静

Student ID:2018533144

E-mail: renyj@shanghaitech.edu.cn

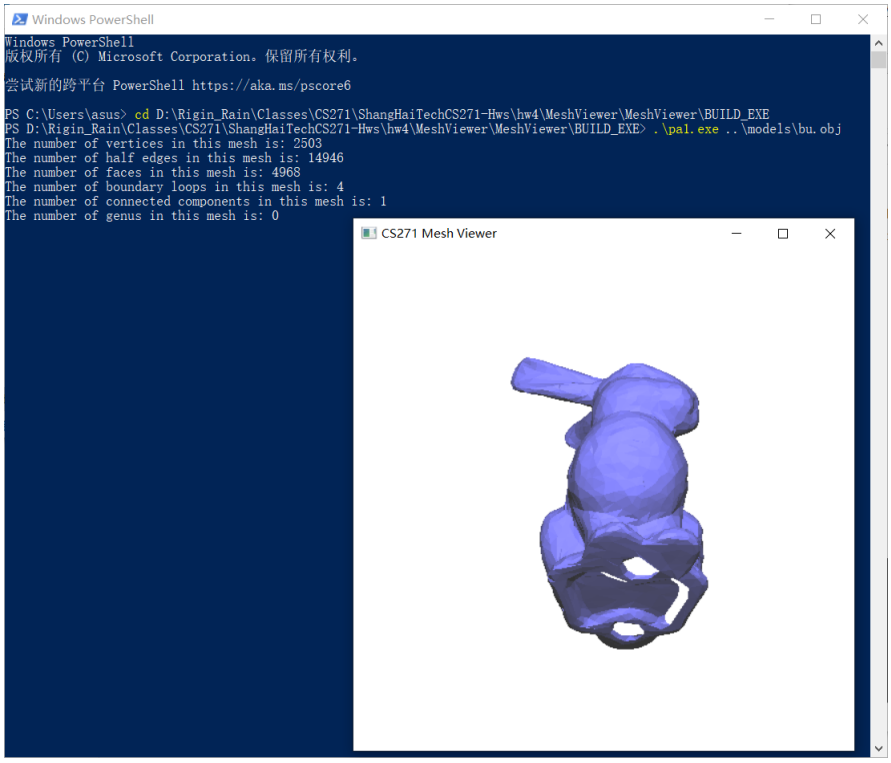
Problem 1: Use Explicit and Implicit Laplacian method with Uniform weights and Cotangent weights, also show the point normals and curvatures

Package and Environment

- The convex hull algorithm is realized using language **C++**
- The project only uses the **GLUT** library as provided.
- Follow the instructions (https://piazza.com/class/kl8wmh1uda82ps?cid=29) on Piazza if **GULT** is not available on your computer

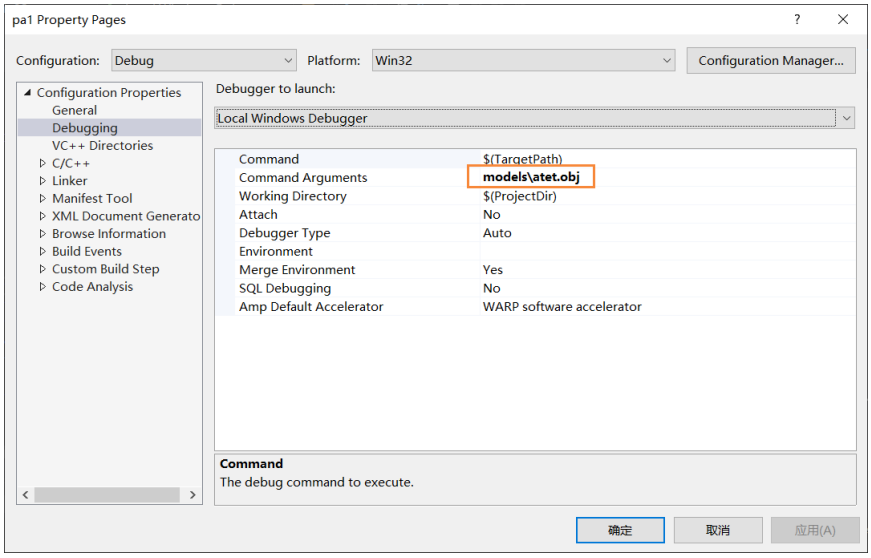
Instructions

- If you use command lines **(Windows Powershell recommended)**
  - First Enter the `BUILD_EXE` folder in the terminal window
  - then run the EXE with `.\pa1.exe ..\models\{THE_OBJ_FILENAME} {USE_UNIFORM_WEIGHT}`
  - (`./pa1.exe ../models/{THE_OBJ_FILENAME}` in Linux based system maybe, I haven't tried)
  - For example
    - if you want to check model **bu.obj** with uniform weight, simply type `.\pa1.exe ..\models\bu.obj m`
    - Or if you want to check model **bu.obj** with cotangent weight, simply type `.\pa1.exe ..\models\bu.obj`



    - **Press "U" on keyboard for explicit smoothing, press "S" on keyboard for implicit smoothing, KEEP PRESSING UNTIL YOU FEEL SATISFIED WITH YOUR RESULT.**

- If you use visual studio (2017) to open the pa1.sln
    - In **Project -> pa1 Properties... -> Debugging**, Set **Command Arguments** to `models\{THE_OBJ_FILENAME} {USE_UNIFORM_WEIGHT}`.



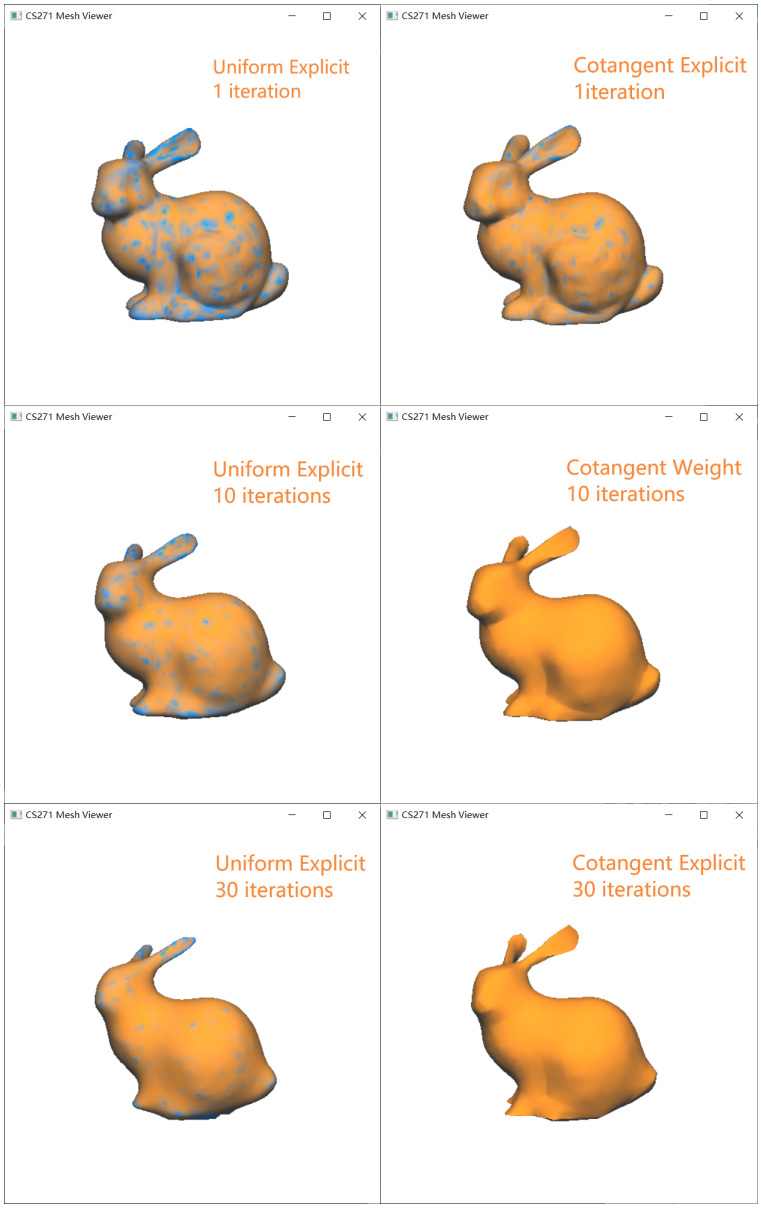
    - For example
        - if you want to check model **bu.obj** with uniform weight, simply write in `models\bu.obj m`
        - Or if you want to check model **bu.obj** with cotangent weight, simply write in `models\bu.obj`
    - **Press "U" on keyboard for explicit smoothing, press "S" on keyboard for implicit smoothing, KEEP PRESSING UNTIL YOU FEEL SATISFIED WITH YOUR RESULT.**

Output

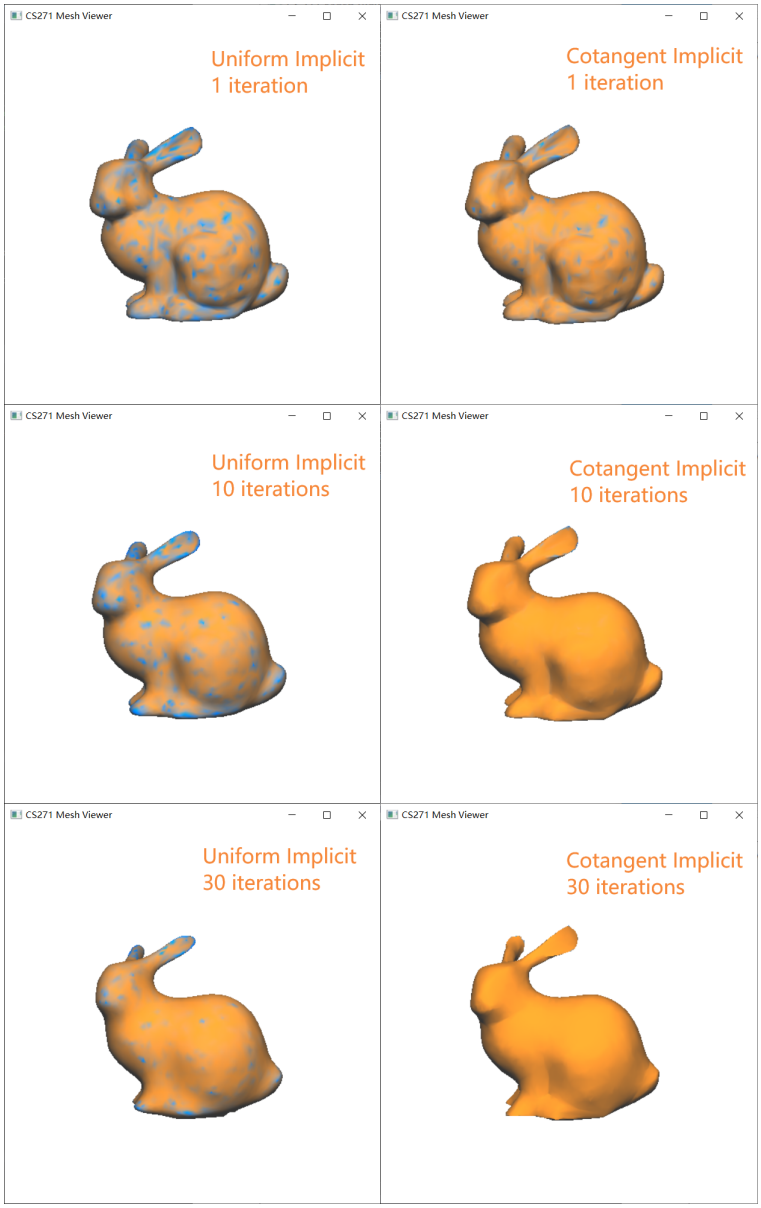
- It will output in console a line of code showing the required type of smoothing is finished.
- It will output a window displaying the mesh, drag left mouse button to rotate and hold on to middle mouse button and drag to scale. The good areas will be in blue, and the areas that are not so good will appear as blue. The curvature and normal are demostrated as the colors on the mesh.

Results(Just picked some obvious models for demostration)

- For bu.obj
  - Explicit Smoothing (Left Uniform weight, right Cotangent weight)



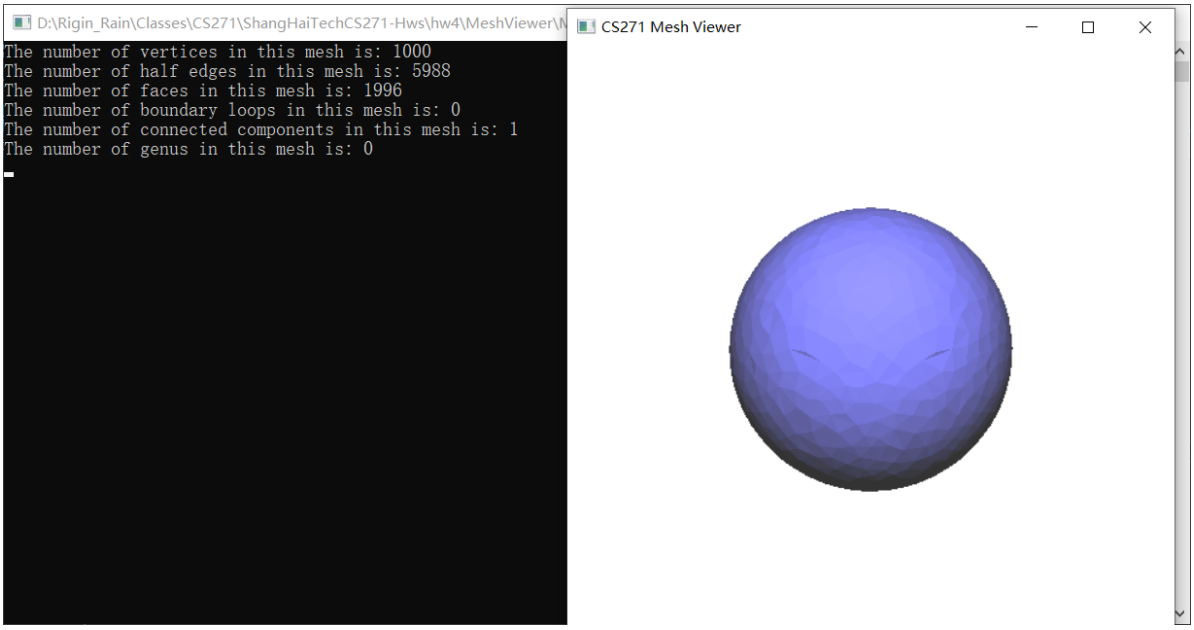
  - Implicit Smoothing (Left Uniform weight, right Cotangent weight)



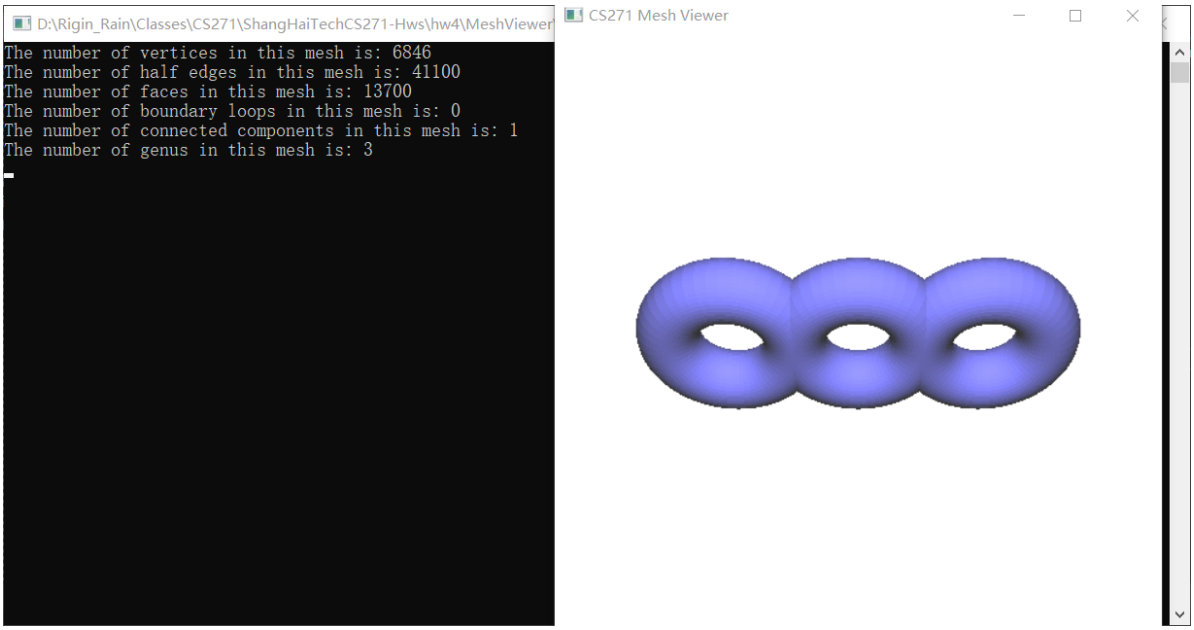
- For cactus.obj

- For gbones.obj

- For sp.obj



- For torus03.obj



Problem 2: Use Explicit and Implicit Laplacian method with Cotangent weights, also show the point normals and curvatures

Package and Environment

- The same as problem 1

Instructions

- Run the code as problem 1

Output

- The display window will show the result in real time.

Notice

- Since there is no maintenance in vertices' Indices, it is not recommended to delete another vertex.
- Due to not maintaining the vertices's Indices, the menu when right click the mouse is disabled if one vertex is deleted
- Sometimes if scale the view too close, the program will provide error, but this seems to be the bug of the template, just restart the program and be careful when scrolling view

Result (Selected Some rough models (less vertices) for better view)

- Performance on ordinary non-bounding vertex **(mannequin.obj)**
    - As the first two figures, you can see the vertex has been deleted and the layout of the face around the deleted vertex changes to fit the rules
- The following two pictures shows that the program prohibits the second deletion, it also prevents some unintended triggering.
- Performance on ordinary bounding vertex **(mannequin.obj)**
    - As in figures, the corresponding vertex has been deleted, you can see the tiny difference in the way the boundary edge goes

- Performance on non-bounding Edge vertex (sharp corners) **(gbones.obj, atet.obj)**
    - As you can see, even if deleting an sharp corner, the mending of the faces still performs good